UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/083,872	02/27/2002	Eric DeLano	10016663-1	4721

7590 11/01/2006

HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, CO 80527-2400

EXAMINER	
LI, AIMEE J	

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 11/01/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/083,872	DELANO, ERIC	
	Examiner	Art Unit	
	Aimee J. Li	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *07 August 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-10,12,13 and 15-18* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-10, 12-13, and 15-18* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-10, 12-13, and 15-18 have been considered.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: Amendment as filed 07 August 2006 and Extension of Time for 1 Month as filed 07 August 2006.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claims 1-10, 12-13, 15-16, and 18 are rejected under 35 U.S.C. 102(e) as being taught by Intel's "Hyper-Threading Technology" in Intel Technology Journal: Volume 06 Issue 01 published 14 February 2002 (herein referred to as Intel).

5. Referring to claim 1, Intel has taught a method for processing bundled instructions through execution units of a processor, comprising the steps of:

a. Determining a throughput mode of operation, based upon a configuration bit (Intel page 12, columns 1-2, Single-Task and Multi-Task Modes all);

b. Fetching a first bundle of singly-threaded instructions from a singly-or multiple-threaded program (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2;

page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6);

c. Distributing the first bundle to a first cluster of the execution units for execution therethrough (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6);

d. Fetching a second bundle of singly-threaded instructions from the program (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6); and

e. Distributing the second bundle to a second cluster of the execution units for execution therethrough (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7,

column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6).

6. Referring to claim 2, Intel has taught processing the first bundle within the first cluster (Intel page 12, columns 1-2, Single-Task and Multi-Task Modes all).

7. Referring to claim 3, Intel has taught processing the second bundle within the second cluster (Intel page 12, columns 1-2, Single-Task and Multi-Task Modes all).

8. Referring to claim 4, Intel has taught architecting data from the first cluster to a first register file (Intel page 7, column 1, paragraph 3 to column 2, paragraph 2).

9. Referring to claim 5, Intel has taught committing architected state from the second cluster to the first register file (Intel page 7, column 1, paragraph 3 to column 2, paragraph 2).

10. Referring to claim 6, Intel has taught architecting data from the second cluster to a second register file (Intel page 7, column 1, paragraph 3 to column 2, paragraph 2).

11. Referring to claim 7, Intel has taught fetching the first bundle comprising decoding instructions into the first bundle of the singly-threaded instructions (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6).

12. Referring to claim 8, Intel has taught fetching the second bundle comprising decoding instructions into the second bundle of the singly-threaded instructions (Intel page 4, column 1,

Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6).

13. Referring to claim 9, Intel has taught

a. Selecting the configuration bit to specify a wide mode of operation (Intel page 12, columns 1-2, Single-Task and Multi-Task Modes all);

b. Fetching a third bundle of singly-threaded instructions from the program (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6);

c. Distributing the third bundle to the first and second clusters of the execution units for execution therethrough (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6); and

d. Bypassing data between the clusters, as needed, to facilitate processing of the third bundle through the clusters (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6).

14. Referring to claim 10, Intel has taught utilizing a latch to couple the data between the clusters (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6).

15. Referring to claim 12, Intel has taught a method for processing bundled instructions through execution units of a processor, comprising the steps of:

a. Determining a wide mode of operation, based upon a configuration bit (Intel page 12, columns 1-2, Single-Task and Multi-Task Modes all);

b. Fetching a first bundle of singly-threaded instructions from a singly- or multiply-threaded program (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel

Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6);

c. Distributing the first bundle to two or more clusters of the execution units for execution therethrough (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6); and

d. Bypassing data between the clusters, as needed, to facilitate processing of the first bundle through the clusters (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6).

16. Referring to claim 13, Intel has taught

a. Selecting the configuration bit to indicate a throughput mode of operation (Intel page 12, columns 1-2, Single-Task and Multi-Task Modes all);

b. Fetching a second bundle of singly-threaded instructions from the program (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level

Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6);

c. Distributing the second bundle to one of the clusters for execution therethrough (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6);

d. Fetching a third bundle of singly-threaded instructions (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6); and

e. Distributing the third bundle to another one of the clusters units for execution therethrough (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1,

paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6).

17. Referring to claim 15, Intel has taught in a processor architecture of the type having two or more clusters of execution units for processing instructions, the improvement comprising:

a. A configuration bit for specifying a wide mode or a throughput mode of operation (Intel page 12, columns 1-2, Single-Task and Multi-Task Modes all);

b. A thread decoder for grouping instructions of a singly- or multiply-threaded program into singly-threaded bundles and for distributing the bundles to the clusters according to the configuration bit (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6; and page 12, columns 1-2, Single-Task and Multi-Task Modes all and Figure 7);

c. Wherein the singly-threaded bundles are distributed across a plurality of clusters in the wide mode and each singly-threaded bundle is distributed to one of the clusters in throughput mode (Intel page 12, columns 1-2, Single-Task and Multi-Task Modes all).

18. Referring to claim 16, Intel has taught wherein each cluster comprises a core and register file (Intel page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6).

19. Referring to claim 18, Intel has taught a method for processing bundled instructions through execution units of a processor, comprising the steps of:

a. Determining, based upon a configuration bit, a throughput mode or wide mode of operation (Intel page 12, columns 1-2, Single-Task and Multi-Task Modes all);

b. Fetching a first bundle of singly-threaded instructions from a singly- or multiply-threaded program (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family, paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6);

c. If in throughput mode of operation, distributing the first bundle to a first cluster of the execution units for execution therethrough (Intel page 12, columns 1-2, Single-Task and Multi-Task Modes all);

d. If in wide mode of operation, distributing the first bundle to mu1tiple clusters of the execution units for execution therethrough (Intel page 12, columns 1-2, Single-Task and Multi-Task Modes all);

e. Fetching a second bundle of singly-threaded instructions from the program (Intel page 4, column 1, Abstract lines 7-14; page 5, Processor Microarchitectures, lines 2-13; page 6, Thread-Level Parallelism, paragraph 1-2; page 6, Thread-Level Parallelism, lines 3-8 and Figure 3; page 7, column 1, paragraph 3 to column 2, paragraph 2; page 7, First Implementation on the Intel Xeon Processor Family,

paragraphs 2-3; and page 10, column 1, Out-of-Order Execution Engine, paragraph 3 and Figure 6);

f. If in throughput mode of operation, distributing the second bundle to a second cluster of the execution units for execution therethrough (Intel page 12, columns 1-2, Single-Task and Multi-Task Modes all); and

g. If in wide mode of operation distributing the second bundle to multiple clusters of the execution units for execution therethrough (Intel page 12, columns 1-2, Single-Task and Multi-Task Modes all).

Response to Arguments

20. Applicant's arguments filed 07 August 2006 have been fully considered but they are not persuasive.

21. Applicants argue in essence on pages 2-3

> Intel instead teaches away from step a), disclosing that "on a processor with HyperThreaded technology, executing HALT transitions the processor from MT-mode to ST0- or ST1-mode, depending on which logical processor executed the HALT."...Again teaching away from step a), Intel discloses that "in ST0- or ST1-modes, an interrupts sent to a HALTed processor would cause a transition to MT-mode."...The mode transition of Intel is not based upon a configuration bit. In fact, Intel does not disclose a configuration bit with relation to single-task and multi-task modes of operation.

22. This has not been found persuasive. It is unclear to the Examiner how the teaching of a HALT transition teaches away from step a) in the claims. Step a) recites "determining a

throughput mode of operation, based upon a configuration bit". The claim language merely states that the operation throughput mode is based on some type of bit. Intel has taught that the HALT interrupt causes a transition. The HALT interrupt is comprised of a number of bits, any of which can be called "a configuration bit", as defined in the claim, since they all cause the throughput mode to change. There is nothing in Intel stating that a configuration bit cannot be used to change the throughput mode. It appears that Applicants arguments are relying on a definition that is not explicitly found in the specification for "a configuration bit" and the Examiner cannot read limitations from the specification into the claim language. In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., a specific meaning to the term "configuration bit") are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

23. Applicant argues in essence on pages 3-7

> ...But, Intel does not disclose or suggest bundling instructions, nor fetching singly-threaded instructions as required by step b). The window of Intel is not equivalent to bundling instructions. Further, Intel does not disclose or suggest distributing bundled instructions to a specific cluster, as required by step c). Intel also does not disclose or suggest fetching a second bundle of singly-threaded instructions, as required by step d), and does not disclose or suggest distributing these second bundle of instructions to a specific cluster, as required by step e). Intel does not disclose bundling of instructions.

24. This has not been found persuasive. Intel has taught two modes of throughput MT-mode and ST0- and ST1-mode (Intel page 12, Single-Task and Multi-Task Modes). MT-mode is multiple thread mode which executes multiple threads throughout the multiple clusters simultaneously. ST0-mode is a single thread mode which executes a single thread through one processor, e.g. cluster 1. ST1-mode is another single thread mode which executes a single thread through another processor, e.g. cluster 2. The distinction between ST0- and ST1-mode means that instructions are designated executed by different clusters, which Intel taught the advantage of in its discussion cited in the above rejection. Intel further teaches the out-of-order execution is utilized to optimize usage of all resources (Intel column 10, column 1, Out-of-Order Execution Engine). Also, Intel has discussed repeatedly throughout the entire document and shown in Figure 5 and Figure 6 that multiple instructions, e.g. instruction bundles, are fetched when in a single mode. Intel has also shown that multiple bundles must be fetched from a single program. For example, Intel on page 13 begins discussion about performance benchmarks of the Xeon™ processor family with regards to 1-4 processors with and without hyper-threading. This means that entire benchmark programs had to have been run utilizing the method described with regards to hyper-threading. Therefore, multiple instructions, e.g. multiple instruction bundles, had to be fetched for execution.

Conclusion

25. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

26. A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO

MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

27. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Aimee J. Li whose telephone number is (571) 272-4169. The examiner can normally be reached on M-T 7:30am-5:00pm.

28. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

29. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

AJL
Aimee J. Li
29 October 2006

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100